Lecture 6: Clustering

Information Retrieval

Computer Science Tripos Part II

Simone Teufel

Natural Language and Information Processing (NLIP) Group

UNIVERSITY OF CAMBRIDGE

Simone.Teufel@cl.cam.ac.uk

Lent 2014
Overview

1 Recap/Catchup

2 Clustering: Introduction

3 Non-hierarchical clustering

4 Hierarchical clustering
### Precision and Recall

**WHAT THE SYSTEM THINKS**

<table>
<thead>
<tr>
<th>Retrieved</th>
<th>Relevant</th>
<th>Nonrelevant</th>
</tr>
</thead>
<tbody>
<tr>
<td>True positives (TP)</td>
<td>false positives (FP)</td>
<td></td>
</tr>
<tr>
<td>Not retrieved</td>
<td>false negatives (FN)</td>
<td>true negatives (TN)</td>
</tr>
</tbody>
</table>

**THE TRUTH**

- True Positives
- False Negatives
- False Positives
- True Negatives

**Formulas**

\[
P = \frac{TP}{TP + FP}
\]

\[
R = \frac{TP}{TP + FN}
\]
Precision/Recall Graph

<table>
<thead>
<tr>
<th>Rank</th>
<th>Doc</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(d_{12})</td>
</tr>
<tr>
<td>2</td>
<td>(d_{123})</td>
</tr>
<tr>
<td>3</td>
<td>(d_4)</td>
</tr>
<tr>
<td>4</td>
<td>(d_{57})</td>
</tr>
<tr>
<td>5</td>
<td>(d_{157})</td>
</tr>
<tr>
<td>6</td>
<td>(d_{222})</td>
</tr>
<tr>
<td>7</td>
<td>(d_{24})</td>
</tr>
<tr>
<td>8</td>
<td>(d_{26})</td>
</tr>
<tr>
<td>9</td>
<td>(d_{77})</td>
</tr>
<tr>
<td>10</td>
<td>(d_{90})</td>
</tr>
</tbody>
</table>
Avg 11pt prec – area under normalised P/R graph

\[ P_{11\text{-}pt} = \frac{1}{11} \sum_{j=0}^{10} \frac{1}{N} \sum_{i=1}^{N} \tilde{P}_i(r_j) \]
Mean Average Precision (MAP)

\[ MAP = \frac{1}{N} \sum_{j=1}^{N} \frac{1}{Q_j} \sum_{i=1}^{Q_j} P(doc_i) \]

**Query 1**

<table>
<thead>
<tr>
<th>Rank</th>
<th>( P(doc_i) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.00</td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.67</td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.50</td>
</tr>
<tr>
<td>7</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0.40</td>
</tr>
<tr>
<td>11</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>0.25</td>
</tr>
</tbody>
</table>

**Query 2**

<table>
<thead>
<tr>
<th>Rank</th>
<th>( P(doc_i) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.00</td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.67</td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.20</td>
</tr>
<tr>
<td>7</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>0.20</td>
</tr>
</tbody>
</table>

\[ MAP = \frac{0.564 + 0.623}{2} = 0.594 \]
What we need for a benchmark

- A collection of **documents**
  - Documents must be representative of the documents we expect to see in reality.
  - There must be many documents.
  - 1398 abstracts (as in Cranfield experiment) no longer sufficient to model modern retrieval

- A collection of **information needs**
  - ...which we will often incorrectly refer to as queries
  - Information needs must be representative of the information needs we expect to see in reality.

- Human **relevance assessments**
  - We need to hire/pay “judges” or assessors to do this.
  - Expensive, time-consuming
  - Judges must be representative of the users we expect to see in reality.
TREC = Text Retrieval Conference (TREC)

Organized by the U.S. National Institute of Standards and Technology (NIST)

TREC is actually a set of several different relevance benchmarks.

Best known: TREC Ad Hoc, used for first 8 TREC evaluations between 1992 and 1999

1.89 million documents, mainly newswire articles, 450 information needs

No exhaustive relevance judgments – too expensive

Rather, NIST assessors’ relevance judgments are available only for the documents that were among the top $k$ returned for some system which was entered in the TREC evaluation for which the information need was developed.
<num> Number: 508
<title> hair loss is a symptom of what diseases
<desc> Description: Find diseases for which hair loss is a symptom.
<narr> Narrative: A document is relevant if it positively connects the loss of head hair in humans with a specific disease. In this context, “thinning hair” and “hair loss” are synonymous. Loss of body and/or facial hair is irrelevant, as is hair loss caused by drug therapy.
Humans decide which document–query pairs are relevant.
### Observation: Judges disagree a lot.

- This means a large impact on absolute performance numbers of each system.
- But virtually no impact on ranking of systems.
- So, the results of information retrieval experiments of this kind can reliably tell us whether system A is better than system B.
- Even if judges disagree.

<table>
<thead>
<tr>
<th>information need</th>
<th>number of docs judged</th>
<th>disagreements</th>
</tr>
</thead>
<tbody>
<tr>
<td>51</td>
<td>211</td>
<td>6</td>
</tr>
<tr>
<td>62</td>
<td>400</td>
<td>157</td>
</tr>
<tr>
<td>67</td>
<td>400</td>
<td>68</td>
</tr>
<tr>
<td>95</td>
<td>400</td>
<td>110</td>
</tr>
<tr>
<td>127</td>
<td>400</td>
<td>106</td>
</tr>
</tbody>
</table>
Example of more recent benchmark: ClueWeb09

- 1 billion web pages
- 25 terabytes (compressed: 5 terabyte)
- Collected January/February 2009
- 10 languages
- Unique URLs: 4,780,950,903 (325 GB uncompressed, 105 GB compressed)
- Total Outlinks: 7,944,351,835 (71 GB uncompressed, 24 GB compressed)
Recall is difficult to measure on the web

Search engines often use precision at top $k$, e.g., $k = 10$ . . .

. . . or use measures that reward you more for getting rank 1 right than for getting rank 10 right.

Search engines also use non-relevance-based measures.

- **Example 1: clickthrough** on first result
  - Not very reliable if you look at a single clickthrough (you may realize after clicking that the summary was misleading and the document is nonrelevant) . . .
  - . . . but pretty reliable in the aggregate.

- **Example 2: A/B testing**
A/B testing

- Purpose: Test a single innovation
- Prerequisite: You have a large search engine up and running.
- Have most users use old system
- Divert a small proportion of traffic (e.g., 1%) to the new system that includes the innovation
- Evaluate with an “automatic” measure like clickthrough on first result
- Now we can directly see if the innovation does improve user happiness.
- Probably the evaluation methodology that large search engines trust most
MRS, Chapter 8
What is clustering?
Applications of clustering in information retrieval
$K$-means algorithm
Introduction to hierarchical clustering
Single-link and complete-link clustering
Overview

1. Recap/Catchup
2. Clustering: Introduction
3. Non-hierarchical clustering
4. Hierarchical clustering
Clustering: Definition

- (Document) clustering is the process of **grouping a set of documents into clusters of similar documents**.
  - Documents within a cluster should be similar.
  - Documents from different clusters should be dissimilar.
- Clustering is the most common form of **unsupervised** learning.
- Unsupervised = there are no labeled or annotated data.
## Difference clustering–classification

<table>
<thead>
<tr>
<th>Classification</th>
<th>Clustering</th>
</tr>
</thead>
<tbody>
<tr>
<td>supervised learning</td>
<td>unsupervised learning</td>
</tr>
<tr>
<td>classes are human-defined and part of the input to the learning algorithm</td>
<td>Clusters are inferred from the data without human input.</td>
</tr>
<tr>
<td>output = membership in class only</td>
<td>Output = membership in class + distance from centroid (“degree of cluster membership”)</td>
</tr>
</tbody>
</table>
The cluster hypothesis

Cluster hypothesis.

Documents in the same cluster behave similarly with respect to relevance to information needs.

All applications of clustering in IR are based (directly or indirectly) on the cluster hypothesis.

Van Rijsbergen’s original wording (1979): “closely associated documents tend to be relevant to the same requests”.
Applications of Clustering

- **IR:** presentation of results (clustering of documents)
- **Summarisation:**
  - clustering of similar documents for multi-document summarisation
  - clustering of similar sentences for re-generation of sentences
- **Topic Segmentation:** clustering of similar paragraphs (adjacent or non-adjacent) for detection of topic structure/importance
- **Lexical semantics:** clustering of words by cooccurrence patterns
Scatter-Gather

Scatter

Education  Domestic  Iraq  Arts  Sports  Oil  Germany  Legal

Gather

International Stories

Scatter

Deployment  Politics  Germany  Pakistan  Africa  Markets  Oil  Hostages

Gather

Smaller International Stories

Scatter

Trinidad  W. Africa  S. Africa  Security  International  Lebanon  Pakistan  Japan
Clustering search results

1. Data Mining - Wikipedia
   Data mining is the process of extracting patterns from data. Data mining can be used to uncover patterns in data but is often carried out only on samples of data.
   [en.wikipedia.org/wiki/Data_minining]

2. Data Mining: Definition from Answers.com
   Data mining involves the process of extracting useful, often unexpected, information from large databases or data.
   [answers.com/topic/answers-what-is-data-mining]

3. An Introduction to Data Mining
   Data mining tools predict future trends and behaviors, allowing businesses to make proactive, knowledge-based decisions. Data mining tools can answer business questions that traditionally could not.

4. Data Mining: What is Data Mining?
   Data mining software is one of a number of analytical tools for analyzing data. Although data mining is a relatively new term, the technology is not.

5. Data Mining
   In data mining, millions of rows of data must be sifted to find a handful of precious... Data mining starts with the raw data, which usually takes the form of...
Clustering news articles

Ukrainian ex-leader Viktor Yanukovych vows fightback

South Korea calls North missile tests calculated provocation

Venezuela student protest in Caracas ends in clashes

Israe urges IAEA to issue full report on Iran nuclear research

S. Africa Prepares for Historic Mandela Trial

India Sahara chief Suriya Raji arrested

Turkey PM 'tapped calls fabricated by the police'

North Korea States its Preserving Carrony

Putin for lurer house arrest

Chinese court rule Putin for lurer house arrest

Russia court rule Putin for lurer house arrest

FIFA police round up 10 on Planet

Oscar 87: UCLA 83 (2OT)

#8Crazy/Ingramas: Dale Jr., dishes on music, BBQ & all those selfish

Tennis: Murray escapes to victory, Ferrer retires in Acapulco

Jason Collins meets with Shephards

Russia court rule Putin for lurer house arrest

Tea Party insists it's alive and kicking

China hits back at US in human rights report

Rory McIlroy takes one-shot lead at Honda Classic

For best picture, it has to be '12 Years a Slave'

Oscars: Where To Watch Around The World

We know how Bitcoin Prohibition would end

Oil Extends Losses

China Yuan Ends at 8-Month Low, Slides Nearly One Percent Earlier

European stocks weaken after inflation data

Mt. Gox Files for Bankruptcy

Indias Growth Rate Under 5%; Central Bank Promotes No Shrink Therapy

Latest airline panic: Safe distances from the tear gas
Clustering topical areas
Clustering what appears in AAG conferences

(AAG = Association of American Geographers)
Clustering patents
Clustering terms
Types of Clustering

- Hard clustering v. soft clustering
  - **Hard** clustering: every object is member in only one cluster
  - **Soft** clustering: objects can be members in more than one cluster

- Hierarchical v. non-hierarchical clustering
  - **Hierarchical** clustering: pairs of most-similar clusters are iteratively linked until all objects are in a clustering relationship
  - **Non-hierarchical** clustering results in flat clusters of “similar” documents
General goal: **put related docs in the same cluster, put unrelated docs in different clusters.**

- We’ll see different ways of formalizing this.

**The number of clusters** should be appropriate for the data set we are clustering.

- Initially, we will assume the number of clusters $K$ is given.
- There also exist semiautomatic methods for determining $K$.

**Secondary goals in clustering**

- Avoid very small and very large clusters
- Define clusters that are easy to explain to the user
- Many others . . .
1 Recap/Catchup

2 Clustering: Introduction

3 Non-hierarchical clustering

4 Hierarchical clustering
Partitional clustering algorithms produce a set of \( k \) non-nested partitions corresponding to \( k \) clusters of \( n \) objects.

Advantage: not necessary to compare each object to each other object, just comparisons of objects – cluster centroids necessary

Optimal partitioning clustering algorithms are \( O(kn) \)

Main algorithm: \( K \)-means
Each cluster $j$ (with $n_j$ elements $x_i$) is represented by its centroid $c_j$, the average vector of the cluster:

$$c_j = \frac{1}{n_j} \sum_{i=1}^{n_j} x_i$$

Measure of cluster quality: minimise mean square distance between elements $x_i$ and nearest centroid $c_j$

$$RSS = \sum_{j=1}^{k} \sum_{x_i \in j} d(\vec{x}_i, \vec{c}_j)^2$$

Distance: Euclidean; length-normalised vectors in VS

We iterate two steps:

- **reassignment**: assign each vector to its closest centroid
- **recomputation**: recompute each centroid as the average of the vectors that were recently assigned to it
Given: a set $s_0 = \vec{x}_1, \ldots, \vec{x}_n \subseteq \mathcal{R}^m$

Given: a distance measure $d : \mathcal{R}^m \times \mathcal{R}^m \to \mathcal{R}$

Given: a function for computing the mean $\mu : \mathcal{P}(\mathcal{R}) \to \mathcal{R}^m$

Select $k$ initial centers $\vec{c}_1, \ldots, \vec{c}_k$

while stopping criterion not true:

$$\sum_{j=1}^{k} \sum_{x_i \in s_j} d(\vec{x}_i, \vec{c}_j)^2 < \epsilon$$ (stopping criterion)

do

for all clusters $s_j$ do (reassignment)

$$c_j := \{\vec{x}_i | \forall \vec{c}_j : d(\vec{x}_i, \vec{c}_j) \leq d(\vec{x}_i, \vec{c}_j)\}$$

end

for all means $\vec{c}_j$ do (centroid recomputation)

$$\vec{c}_j := \mu(s_j)$$

end
Exercise: (i) Guess what the optimal clustering into two clusters is in this case; (ii) compute the centroids of the clusters
Random seeds + Assign points to closest center

Iteration One
Worked Example: Recompute cluster centroids

Iteration One
Worked Example: Assign points to closest centroid

Iteration One
Worked Example: Recompute cluster centroids

Iteration Two
Worked Example: Assign points to closest centroid

Iteration Two
Worked Example: Recompute cluster centroids

Iteration Three
Worked Example: Assign points to closest centroid

Iteration Three
Worked Example: Recompute cluster centroids

Iteration Four
Worked Example: Assign points to closest centroid

Iteration Four
Worked Example: Recompute cluster centroids

Iteration Five
Worked Example: Assign points to closest centroid

Iteration Five
Worked Example: Recompute cluster centroids

Iteration Six
Worked Example: Assign points to closest centroid

Iteration Six
Worked Example: Recompute cluster centroids

Iteration Seven
Worked Ex.: Centroids and assignments after convergence

Convergence
$K$-means is guaranteed to converge: Proof

- RSS decreases during each reassignment step.
  - because each vector is moved to a closer centroid
- RSS decreases during each recomputation step.
  - This follows from the definition of a centroid: the new centroid is the vector for which $\text{RSS}_k$ reaches its minimum
- There is only a finite number of clusterings.
- Thus: We must reach a fixed point.
- Finite set & monotonically decreasing evaluation function $\rightarrow$ convergence
- Assumption: Ties are broken consistently.
Other properties of $K$-means

- **Fast convergence**
  - $K$-means typically converges in around 10-20 iterations (if we don’t care about a few documents switching back and forth)
  - However, complete convergence can take many more iterations.

- **Non-optimality**
  - $K$-means is not guaranteed to find the optimal solution.
  - If we start with a bad set of seeds, the resulting clustering can be horrible.

- **Dependence on initial centroids**
  - Solution 1: Use $i$ clusterings, choose one with lowest RSS
  - Solution 2: Use prior hierarchical clustering step to find seeds with good coverage of document space
Time complexity of $K$-means

- Reassignment step: $O(KNM)$ (we need to compute $KN$ document-centroid distances, each of which costs $O(M)$)
- Recomputation step: $O(NM)$ (we need to add each of the document’s $< M$ values to one of the centroids)
- Assume number of iterations bounded by $I$
- Overall complexity: $O(IKNM)$ – linear in all important dimensions
Overview

1 Recap/Catchup

2 Clustering: Introduction

3 Non-hierarchical clustering

4 Hierarchical clustering
Imagine we now want to create a hierarchy in the form of a binary tree.
Assumes a similarity measure for determining the similarity of two clusters.
Up to now, our similarity measures were for documents.
We will look at different cluster similarity measures.
Main algorithm: HAC (hierarchical agglomerative clustering)
HAC: Basic algorithm

- Start with each document in a separate cluster
- Then repeatedly merge the two clusters that are most similar
- Until there is only one cluster.
- The history of merging is a hierarchy in the form of a binary tree.
- The standard way of depicting this history is a dendrogram.
NYSE closing averages
Hog prices tumble
Oil prices slip
Ag trade reform.
Chrysler / Latin America
Japanese prime minister / Mexico
Fed holds interest rates steady
Fed to keep interest rates steady
Fed keeps interest rates steady
Fed keeps interest rates steady
Mexican markets
British FTSE index
War hero Colin Powell
War hero Colin Powell
Lloyd’s CEO questioned
Lloyd’s chief / U.S. grilling
Ohio Blue Cross
Lawsuit against tobacco companies
suits against tobacco firms
Indiana tobacco lawsuit
Viag stays positive
Most active stocks
CompuServe reports loss
Sprint / Internet access service
Planet Hollywood
Trocadero: tripling of revenues
Back–to–school spending is up
German unions split
Chains may raise prices
Clinton signs law
Applying the proximity metric to all pairs of documents...

creates the document-document matrix, which reports similarities/distances between objects (documents)

The diagonal is trivial (identity)

As proximity measures are symmetric, the matrix is a triangle
Hierarchical clustering: agglomerative (BottomUp, greedy)

Given: a set \( X = x_1, \ldots, x_n \) of objects;
Given: a function \( \text{sim} : \mathcal{P}(X) \times \mathcal{P}(X) \to \mathcal{R} \)

\[
\begin{align*}
\text{for } i &:= 1 \text{ to } n \text{ do} \\
& \quad c_i := x_i \\
C &:= c_1, \ldots, c_n \\
j &:= n+1 \\
\text{while } C > 1 \text{ do} \\
& \quad (c_{n_1}, c_{n_2}) := \max_{(c_u, c_v) \in C \times C} \text{sim}(c_u, c_v) \\
& \quad c_j := c_{n_1} \cup c_{n_2} \\
& \quad C := C \setminus \{c_{n_1}, c_{n_2}\} \cup c_j \\
j &:= j+1 \\
\end{align*}
\]

Similarity function \( \text{sim} : \mathcal{P}(X) \times \mathcal{P}(X) \to \mathcal{R} \) measures similarity between clusters, not objects
First, we compute the similarity of all $N \times N$ pairs of documents.

Then, in each of $N$ iterations:

- We scan the $O(N \times N)$ similarities to find the maximum similarity.
- We merge the two clusters with maximum similarity.
- We compute the similarity of the new cluster with all other (surviving) clusters.

There are $O(N)$ iterations, each performing a $O(N \times N)$ “scan” operation.

Overall complexity is $O(N^3)$.

Depending on the similarity function, a more efficient algorithm is possible.
Hierarchical clustering: similarity functions

Similarity between two clusters $c_k$ and $c_j$ (with similarity measure $s$) can be interpreted in different ways:

- **Single Link Function**: Similarity of two most similar members
  \[ \text{sim}(c_u, c_v) = \max_{x \in c_u, y \in c_k} s(x, y) \]

- **Complete Link Function**: Similarity of two least similar members
  \[ \text{sim}(c_u, c_v) = \min_{x \in c_u, y \in c_k} s(x, y) \]

- **Group Average Function**: Avg. similarity of each pair of group members
  \[ \text{sim}(c_u, c_v) = \text{avg}_{x \in c_u, y \in c_k} s(x, y) \]
Example: hierarchical clustering; similarity functions

Cluster 8 objects a-h; Euclidean distances (2D) shown in diagram

\[
\begin{array}{cccc}
  & a & b & c \\
 b & 1 & & \\
 c & 2.5 & 1.5 & \\
 d & 3.5 & 2.5 & 1 \\
 e & 2 & \sqrt{5} & \sqrt{10.25} & \sqrt{16.25} \\
 f & \sqrt{5} & 2 & \sqrt{6.25} & \sqrt{10.25} & 1 \\
 g & \sqrt{10.25} & \sqrt{6.25} & 2 & \sqrt{5} & 2.5 & 1.5 \\
 h & \sqrt{16.25} & \sqrt{10.25} & \sqrt{5} & 2 & 3.5 & 2.5 & 1 \\
\end{array}
\]
Single Link is $O(n^2)$

After Step 4 (a–b, c–d, e–f, g–h merged):

```
<table>
<thead>
<tr>
<th></th>
<th>c–d</th>
<th>1.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>e–f</td>
<td>2</td>
<td>$\sqrt{6.25}$</td>
</tr>
<tr>
<td>g–h</td>
<td>$\sqrt{6.25}$</td>
<td>2</td>
</tr>
<tr>
<td>a–b</td>
<td>c–d</td>
<td>e–f</td>
</tr>
</tbody>
</table>
```

“min-min” at each step
Clustering Result under Single Link

\[
\begin{array}{cccc}
  a & b & c & d \\
  e & f & g & h \\
\end{array}
\]
After step 4 (a–b, c–d, e–f, g–h merged):

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>g</th>
</tr>
</thead>
<tbody>
<tr>
<td>a–b</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>c–d</td>
<td>2.5</td>
<td>1.5</td>
<td>3.5</td>
<td>2.5</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>e–f</td>
<td>2</td>
<td></td>
<td>2</td>
<td></td>
<td></td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>g–h</td>
<td>2</td>
<td></td>
<td>3.5</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

“max-min” at each step
<table>
<thead>
<tr>
<th></th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>g</th>
<th>h</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>1</td>
<td>2.5</td>
<td>3.5</td>
<td>2</td>
<td>$\sqrt{5}$</td>
<td>$\sqrt{10.25}$</td>
<td>$\sqrt{16.25}$</td>
</tr>
<tr>
<td>b</td>
<td>2</td>
<td>$\sqrt{5}$</td>
<td>2</td>
<td>$\sqrt{6.25}$</td>
<td>$\sqrt{10.25}$</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>1.5</td>
<td>$\sqrt{10.25}$</td>
<td>2</td>
<td>$\sqrt{5}$</td>
<td>2.5</td>
<td>1.5</td>
<td></td>
</tr>
<tr>
<td>d</td>
<td>1</td>
<td>$\sqrt{16.25}$</td>
<td>$\sqrt{10.25}$</td>
<td>$\sqrt{5}$</td>
<td>2</td>
<td>2.5</td>
<td>1</td>
</tr>
<tr>
<td>e</td>
<td>1.5</td>
<td>2.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

After step 4 (a–b, c–d, e–f, g–h merged):

<table>
<thead>
<tr>
<th>c–d</th>
<th>2.5</th>
<th>1.5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3.5</td>
<td>2.5</td>
</tr>
<tr>
<td>e–f</td>
<td>2</td>
<td>$\sqrt{5}$</td>
</tr>
<tr>
<td></td>
<td>$\sqrt{5}$</td>
<td>2</td>
</tr>
<tr>
<td>g–h</td>
<td>$\sqrt{10.25}$</td>
<td>$\sqrt{6.25}$</td>
</tr>
<tr>
<td></td>
<td>$\sqrt{16.25}$</td>
<td>$\sqrt{10.25}$</td>
</tr>
</tbody>
</table>

“max-min” at each step $\rightarrow$ ab/ef and cd/gh merges next
Clustering result under complete link

Complete Link is $O(n^3)$
Example: gene expression data

- An example from biology: cluster genes by function
- Survey 112 rat genes which are suspected to participate in development of CNS
- Take 9 data points: 5 embryonic (E11, E13, E15, E18, E21), 3 postnatal (P0, P7, P14) and one adult
- Measure expression of gene (how much mRNA in cell?)
- These measures are normalised logs; for our purposes, we can consider them as weights
- Cluster analysis determines which genes operate at the same time
<table>
<thead>
<tr>
<th>gene</th>
<th>genbank locus</th>
<th>E11</th>
<th>E13</th>
<th>E15</th>
<th>E18</th>
<th>E21</th>
<th>P0</th>
<th>P7</th>
<th>P14</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>keratin</td>
<td>RNKER19</td>
<td>1.703</td>
<td>0.349</td>
<td>0.523</td>
<td>0.408</td>
<td>0.683</td>
<td>0.461</td>
<td>0.32</td>
<td>0.081</td>
<td>0</td>
</tr>
<tr>
<td>cellubrevin</td>
<td>s63830</td>
<td>5.759</td>
<td>4.41</td>
<td>1.195</td>
<td>2.134</td>
<td>2.306</td>
<td>2.539</td>
<td>3.892</td>
<td>3.953</td>
<td>2.72</td>
</tr>
<tr>
<td>nestin</td>
<td>RATNESTIN</td>
<td>2.537</td>
<td>3.279</td>
<td>5.202</td>
<td>2.807</td>
<td>1.5</td>
<td>1.12</td>
<td>0.532</td>
<td>0.514</td>
<td>0.433</td>
</tr>
<tr>
<td>MAP2</td>
<td>RATMAP2</td>
<td>0.04</td>
<td>0.514</td>
<td>1.553</td>
<td>1.654</td>
<td>1.66</td>
<td>1.491</td>
<td>1.436</td>
<td>1.585</td>
<td>1.894</td>
</tr>
<tr>
<td>GAP43</td>
<td>RATGAP43</td>
<td>0.874</td>
<td>1.494</td>
<td>1.677</td>
<td>1.937</td>
<td>2.322</td>
<td>2.296</td>
<td>1.86</td>
<td>1.873</td>
<td>2.396</td>
</tr>
<tr>
<td>L1</td>
<td>S55536</td>
<td>0.062</td>
<td>0.162</td>
<td>0.51</td>
<td>0.929</td>
<td>0.966</td>
<td>0.867</td>
<td>0.493</td>
<td>0.401</td>
<td>0.384</td>
</tr>
<tr>
<td>NFM</td>
<td>RATNFM</td>
<td>0.571</td>
<td>3.373</td>
<td>5.155</td>
<td>4.092</td>
<td>4.542</td>
<td>7.03</td>
<td>6.682</td>
<td>13.591</td>
<td>27.692</td>
</tr>
<tr>
<td>NFH</td>
<td>RATNFHPEP</td>
<td>0.166</td>
<td>0.141</td>
<td>0.545</td>
<td>1.141</td>
<td>1.553</td>
<td>1.667</td>
<td>1.929</td>
<td>4.058</td>
<td>3.859</td>
</tr>
<tr>
<td>synaptophysin</td>
<td>RNSYN</td>
<td>0.205</td>
<td>0.636</td>
<td>1.571</td>
<td>1.476</td>
<td>1.948</td>
<td>2.005</td>
<td>2.381</td>
<td>2.191</td>
<td>1.757</td>
</tr>
<tr>
<td>neno</td>
<td>RATENONS</td>
<td>0.27</td>
<td>0.704</td>
<td>1.419</td>
<td>1.469</td>
<td>1.861</td>
<td>1.556</td>
<td>1.639</td>
<td>1.586</td>
<td>1.512</td>
</tr>
<tr>
<td>S100 beta</td>
<td>RATS100B</td>
<td>0.052</td>
<td>0.011</td>
<td>0.491</td>
<td>1.303</td>
<td>1.487</td>
<td>1.357</td>
<td>1.438</td>
<td>2.275</td>
<td>2.169</td>
</tr>
<tr>
<td>GFAP</td>
<td>RNU03700</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.292</td>
<td>2.705</td>
<td>3.731</td>
<td>8.705</td>
<td>7.453</td>
<td>6.547</td>
</tr>
<tr>
<td>MOG</td>
<td>RATMOG</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.012</td>
<td>0.385</td>
<td>1.462</td>
<td>2.08</td>
<td>1.816</td>
</tr>
<tr>
<td>GAD65</td>
<td>RATGAD65</td>
<td>0.353</td>
<td>1.117</td>
<td>2.539</td>
<td>3.808</td>
<td>3.212</td>
<td>2.792</td>
<td>2.671</td>
<td>2.327</td>
<td>2.351</td>
</tr>
<tr>
<td>pre-GAD67</td>
<td>RATGAD67</td>
<td>0.073</td>
<td>0.18</td>
<td>1.171</td>
<td>1.436</td>
<td>1.443</td>
<td>1.383</td>
<td>1.164</td>
<td>1.003</td>
<td>0.985</td>
</tr>
<tr>
<td>GAD67</td>
<td>RATGAD67</td>
<td>0.297</td>
<td>0.307</td>
<td>1.066</td>
<td>2.796</td>
<td>3.572</td>
<td>3.182</td>
<td>2.604</td>
<td>2.307</td>
<td>2.079</td>
</tr>
<tr>
<td>G67I80/86</td>
<td>RATGAD67</td>
<td>0.767</td>
<td>1.38</td>
<td>2.35</td>
<td>1.88</td>
<td>1.332</td>
<td>1.002</td>
<td>0.668</td>
<td>0.567</td>
<td>0.304</td>
</tr>
<tr>
<td>G67I86</td>
<td>RATGAD67</td>
<td>0.071</td>
<td>0.204</td>
<td>0.641</td>
<td>0.764</td>
<td>0.406</td>
<td>0.202</td>
<td>0.052</td>
<td>0.022</td>
<td>0</td>
</tr>
<tr>
<td>GAT1</td>
<td>RATGABAT</td>
<td>0.839</td>
<td>1.071</td>
<td>5.687</td>
<td>3.864</td>
<td>4.786</td>
<td>4.701</td>
<td>4.879</td>
<td>4.601</td>
<td>4.679</td>
</tr>
<tr>
<td>ChAT</td>
<td>(*)</td>
<td>0</td>
<td>0.022</td>
<td>0.369</td>
<td>0.322</td>
<td>0.663</td>
<td>0.597</td>
<td>0.795</td>
<td>1.015</td>
<td>1.424</td>
</tr>
<tr>
<td>ACHE</td>
<td>S50879</td>
<td>0.174</td>
<td>0.425</td>
<td>1.63</td>
<td>2.724</td>
<td>3.279</td>
<td>3.519</td>
<td>4.21</td>
<td>3.885</td>
<td>3.95</td>
</tr>
<tr>
<td>ODC</td>
<td>RATODC</td>
<td>1.843</td>
<td>2.003</td>
<td>1.803</td>
<td>1.618</td>
<td>1.569</td>
<td>1.565</td>
<td>1.394</td>
<td>1.314</td>
<td>1.11</td>
</tr>
<tr>
<td>TH</td>
<td>RATTOHA</td>
<td>0.633</td>
<td>1.225</td>
<td>1.007</td>
<td>0.801</td>
<td>0.654</td>
<td>0.691</td>
<td>0.23</td>
<td>0.287</td>
<td>0</td>
</tr>
<tr>
<td>NOS</td>
<td>RRBNOS</td>
<td>0.051</td>
<td>0.141</td>
<td>0.675</td>
<td>0.63</td>
<td>0.86</td>
<td>0.926</td>
<td>0.792</td>
<td>0.646</td>
<td>0.448</td>
</tr>
<tr>
<td>GRa1</td>
<td>(#)</td>
<td>0.454</td>
<td>0.626</td>
<td>0.802</td>
<td>0.972</td>
<td>1.021</td>
<td>1.182</td>
<td>1.297</td>
<td>1.469</td>
<td>1.511</td>
</tr>
</tbody>
</table>
Rat CNS gene clustering – single link

Clustering of Rat Expression Data (Single Link/Euclidean)
Rat CNS gene clustering – group average link

Clustering of Rat Expression Data (Av Link/Euclidean)

0 20 40 60 80 100

Gene symbols include:
- Chromatin-associated proteins: H2AZ, Statin, c-Jun, c-Fos, Brm, TCP
- Neurotransmitter receptors: mGluR1, mGluR2, mGluR3, mGluR4, mGluR5, mGluR6, mGluR7, mGluR8, NMDA1, NMDA2A, NMDA2B, NMDA2C, NMDA2D
- Acetylcholine receptors: nAChRa2, nAChRa3, nAChRa4, nAChRa5, nAChRa6, nAChRa7
- Neurotrophic factors and their receptors: NGF, NT3, BDNF, CNTF, trk, trkB, trkC, CNTFR, IGFR1, IGFR2, trkB, trkC, CNTFR
- Neurophins: GDNF, EGF, bFGF, aFGF, PDG Fa, PDGFb
- Growth factors and receptors: IGF I, IGF II, InsR, InsR1, InsR2, InsR3, EGFR, FGFR, PDGFR, TGFR
- Signal transduction proteins: CRAF, IP3R1, IP3R2, IP3R3, cyclin A, cyclin B, cyclin C, CCA2, CCA1, GADD34.2, 0 20 40 60 80 100, 324
Flat or hierarchical clustering?

- When a hierarchical structure is desired: hierarchical algorithm
- Humans are bad at interpreting hierarchical clusterings (unless cleverly visualised)
- For high efficiency, use flat clustering
- For deterministic results, use HAC
- HAC also can be applied if $K$ cannot be predetermined (can start without knowing $K$)
- Partitional clustering
  - Provides less information but is more efficient (best: $O(kn)$)
  - $K$-means

- Hierarchical clustering
  - Best algorithms $O(n^2)$ complexity
  - Single-link vs. complete-link (vs. group-average)

- Hierarchical and non-hierarchical clustering fulfills different needs
Reading

- MRS Chapters 16.1-16.4
- MRS Chapters 17.1-17.2